June 15, 2001
Ref. No.: EOS/ETS-061501-C11

National Aeronautics and
Space Administration
Goddard Space Flight Center
Greenbelt, Maryland 20771

Attention: Mr. Willie Fuller
               Code 581
               Building 32, Room N212D

Subject: Contract No.: NAS9-98100
               CSOC SODA Task Order Number GM36
               EOSDIS Test System (ETS) Multimode Portable Simulator for Aura
               (MPS/Aura) Delivery of the Release 2.0 Software

Dear Mr. Fuller:

We are pleased to deliver Release 2.0 of the ETS Multimode Portable Simulator (MPS) for
the Aura spacecraft (MPS/Aura). This is the second major delivery of MPS/Aura using our
Scalable, Integrated Multimission Simulation Suite (SIMSS) infrastructure and architecture
with extensions for the EOS Aura spacecraft.

The major capabilities in this release consist of emulation of TES segmented telemetry
packets, project database ingest enhancements, conditional scenario script processing, APID
packet status display and resolution of 6 DRs. A complete list of the simulator capabilities is
included in Attachment A. The simulator is set up to use the Aqua Project Data Base (PDB)
for test or demonstration purposes, since an Aura PDB is not yet available.

A hard copy of the MPS/Aura User's Guide for Release 2.0 will be made available to the
users in the near future. In addition, a soft copy of the User's Guide will be placed on the
ETS Documentation web site at URL http://esdis-it.gsfc.nasa.gov/ETS/etsdoc.html.

This delivery package contains 12 attachments as listed below. A completed Mission
Systems Configuration Management (MSCM) form is included in Attachment L. If you have
any questions concerning this delivery, please call me at 301-805-3420.

                              Sincerely yours,

                              Dave Green,
                              CSOC ETS Task Leader

EOSDIS Test System (ETS) Multimode Portable Simulator for Aura (MPS/Aura)
Delivery of the Release 2.0 Software
June 15, 2001, Ref. No.: EOS/ETS-061501-C11

Delivery Package Reviewed and Approved by:

Laura Hepfer
CSOC Quality Assurance

Estelle Noone
CSOC ETS Customer Service Representative

The following attachments contain the details of the MPS software delivery.

Attachment A - describes the delivery contents for this release
Attachment B - describes the operational changes
Attachment C - contains the instructions to build and install the software
Attachment D - contains any special operating instructions
Attachment E - contains a list of the resolved DRs
Attachment F - contains a list of the unresolved DRs
Attachment G - contains the matrix of requirements addressed by this release
Attachment H - contains the known system limitations
Attachment I - contains the release history summary matrix
Attachment J - contains a listing of the delivery media contents
Attachment K - contains documentation references
Attachment L - contains the Mission Systems Configuration Management (MSCM) form

Distribution: (* - Letter Only)

NASA	**Honeywell**	**Lockheed Martin**	**Raytheon**	**CSC**
Caulfield, M. *	Bradbury, T. *	Cordier, G.	Gradishar, G.	Burrows, P.
Johns, A. *	Luo, C. *	Garner, G.	Johnson, W.	Carlson, J.
Kelly, A.		Johnson, T.	Klem, K.	Green, D.
Krimchansky, A.*	**CRC**		Ruland, V.	Hepfer, L.
Lehtonen, K. *	Lavery, K.			Noone, E.
Ondrus, P. *			**Averstar**	Nguyen, T.
Perkins, D. *	**Unisys**	**General-Dynamics**	Messerly, R.	Quintin, E.
ESDIS Library	Thompson, S.	Patterer, R.	Shein, H.	Ramey, D.
			Winters, R.	Shurie, E.
				Xu, F.
				Task File

Attachment A – Description of Delivery Contents

The MPS/Aura Release 2.0 consists of custom software executables that are being delivered on one CD-ROM.   Two copies are being provided.

Enterprise Oracle and Oracle Programmer for Windows NT are necessary for operation of this release. These products were provided with a previous release. The license to use Oracle belongs to the CSOC contract. Therefore, Oracle and Oracle Programmer may be installed only on CSOC computers.

A soft copy of this MPS/Aura Release 2.0 delivery letter and set of attachments is also being delivered. The attachments have been formatted on a 3.5" IBM PC diskette utilizing the MS WORD word processing tool.

Attachment B – Summary of Operational Changes

Operational Capabilities of MPS/Aura Release 2.0

New or modified capabilities with this release are noted in *Italics*.

Since the Aura PDB is not yet available the software is set up to access the Aqua PDB.

Telemetry:

- Transmit telemetry in IP or Serial (clock/data) mode
- Pack telemetry packets and CLCWs into CADUs when in Serial mode
- Generate one stream of CADUs when in Serial mode
- Generate one stream of telemetry formatted as EDUs when in IP mode
- Start or stop one telemetry stream
- Ingest the PDB files
- Generate telemetry packets from information contained in the PDB
- Maintain telemetry nodes from information contained in the PDB
- Populate telemetry packets with data values from information contained in the PDB
- Generate correct secondary headers for SC, GIRD, and SUROM-TIE (no secondary header) telemetry packets using information from the PDB
- Generate instrument telemetry packets using secondary key information from the PDB
- Display EDU data when in IP mode
- Display CADU data when in Serial mode
- Set values into telemetry points by mnemonic
- Display telemetry node values by mnemonic
- Convert telemetry values to Engineering Units (EU) for display using information from the PDB
- Accept operator-entered telemetry values in EU and convert to Raw Counts for inclusion in telemetry packets
- Reset packet count for the telemetry stream
- Static packet data can be overwritten (by byte location) and by modification of telemetry mnemonic
- Incrementing packet sequence counters per APID
- Generation of individual APIDs can be inhibited
- Telemetry logs will be created (viewable by offline utility)
- Packet Headers and Packet Data are updated
- Packet data can be shown in hexadecimal or octal format and addressed in hexadecimal or decimal form
- Packet Sequence Counters can be reset
- Packet Sequence Counters can be modified
- Packet Version field can be modified
- Packet APID field can be modified
- Packet Type field can be modified

- Packet Secondary Header Flag field can be modified
- Packet Length field can be modified
- CCSDS Unsegmented TimeCode (CUC) can be modified
- Packet rate may be controlled
- CLCW transmitted via EDUs when in IP mode
- IP packets are transmitted with variable lengths
- CLCW can be overridden by the operator
- Transmission of CLCW can be inhibited when in IP mode
- Scenario file (script) capability to set telemetry nodes and buffers
- Set telemetry data values in response to spacecraft commands received (end-item verification)
- Set initial telemetry data values at initialization
- Allow simultaneous display and set of multiple telemetry container items via GUI screens
- Simulate spacecraft memory dumps
- Use the PDB telemetry state text file to locate end-item verifier values
- Maintain and update telemetry data values in APID 1000
- Telemetry parameters may be set and viewed by Parameter ID
- CLCW Transmit Start and Stop is coupled to H/K Telemetry Start and Stop
- Telemetry values may be set using simple expressions
- Telemetry values may be set using trigonometric expressions
- Telemetry values may be set using Boolean expressions
- Telemetry values may be set to other telemetry mnemonic values
- Telemetry values may be saved in intermediate variables for later use
- *TES Segmented Packets are emulated*
- *CLCW Transmit rate may be set by the operator*
- *Telemetry data values are validated for fit into packet space*
- *Current enable status and transmit rate for all APIDs is viewable via status display*
- *vcProcessor module discards VC63 VCDUs when creating files for playback*

Command:

- Ingest command-related PDB files
- Identify commands using information from the PDB
- Display event messages with command mnemonics and submnemonics
- Set telemetry points in response to commands received (end-item verification) using information from the PDB
- Recognize spacecraft Command Loads
- Display Command Load data
- Copy Command Load data to a Memory Dump buffer
- Inhibit the Command Load data copy facility via operator directive
- Validate checksums of received Command Loads
- Ingest type AD, BC, and BD commands

- Display Total CLTUs count
- Reset Total CLTUs count
- Display Rejected CLTUs count
- Reset Rejected CLTUs count
- Display Instrument commands count
- Reset Instrument commands count
- Display Spacecraft commands count
- Reset Spacecraft commands count
- Display BC commands count
- Reset BC commands count
- Display BD commands count
- Display current Spacecraft CLCW
- Update Spacecraft and instrument CLCW
- Display current Instrument CLCW
- Validate commands based on individual, all, or none of the following validation criteria: CLTU Start and Tail Sequences, BCH Error Code, Transfer Frame Header Fields, FARM (Valid Frame Sequence), User Command Packet Header
- Generate event messages based on ingest
- Log raw commands (viewable by offline utility)
- Display raw command in hexadecimal or octal format addressed in either hexadecimal or decimal fashion
- Display command packet headers for instrument commands
- Display command packet headers for spacecraft commands
- Update command accepted and rejected counters in telemetry
- Command submnemonics are saved in container items and may be viewed after command receipt
- Expected Spacecraft ID changed to CC Hex

Time:

- Maintain and update SC time (GIRD)
- Maintain and update GMT time
- Synchronize SC and GMT times

General:

– Control all simulator module functions via scenario scripts
– Selection of scenario scripts may be via operator type-in or via a file selection browse window
– Start scenario scripts in response to commands received
– Start a scenario script from a scenario script
– Execute multiple scenario scripts simultaneously
– Provide operator control of multiple scenario scripts started by the operator
– Save the last 10 operator directives

- Allow editing of saved operator directives before re-execution
- EDOS Service Header (ESH) fields may be viewed
- ESH field contents may be modified by the operator
- Validation of Command Data Block (CDB) header fields of commands received
- Modification of expected values of CDB header fields
- All viewable buffers may be displayed
- Addition, deletion, and modification of command end-item verifiers via SQL scripts
- Logs of commands received or telemetry transmitted may be retransmitted via IP output or Serial output
- Expected Spacecraft ID may be modified in EOSGS module
- CLCW ESH field contents may be modified by the operator
- Event messages to the screen may be inhibited or enabled by severity (color)
- *Scenario scripts may contain IF-then-ELSE-ENDIF and WHILE-ENDWHILE conditional execution directives*
- *The Scenario module may interface with multiple modules*
- *Intermediate variables A – Z permit saving values as real numbers*
- *Intermediate variables Aq – Zq permit saving values as long integers*

Attachment C – MPS/Aura Release 2.0 Installation Instructions

This attachment contains the instructions for installing the MPS/Aura Release 2.0 Server and Client. The information presented in this attachment has been checked for accuracy by the independent test team.

Instructions for installation of the Aura Server and Client software:

1. Insert the delivery media into the appropriate drive.

2. To install the Aura Client:
    a) On the desktop, click on the Start button, and then select Run from the resulting menu.
    b) When the Run window appears select the Browse… button.
    c) From the Browse Window, select the Removable drive that contains the installation CD.
    d) Click on the Client folder.
    e) From within the Client folder, double click on the Setup.exe icon.
    f) The screen will be filled with an Aura Client background and a smaller window with the title "Welcome to Aura Client 2.0" will appear. Click on the Next button to proceed to the next step.
    g) The next window will contain the licensing agreement. Click on Yes to accept the agreement and proceed.
    h) After all of the files are copied, a window with the title "Setup Complete" will appear. Click on the Finish button to end.
    i) An Aura Client icon will now be installed on the desktop.

3. To install the Aura Server:
    a) On the desktop, click on the Start button, and then select Run from the resulting menu.
    b) When the Run window appears select the Browse… button.
    c) From the Browse Window, select the Removable drive that contains the installation CD.
    d) Click on the Server folder.
    e) From within the Server folder, double click on the Setup.exe icon
    f) A window with the title "Run Window" will appear. Click on the Okay button to proceed to the next step.
    g) The screen will then be filled with an Aura Server background and a window with the title of "Welcome to Aura Server 2.0" will appear. Click the Next button to proceed.
    h) The next window will contain the licensing agreement. Click on Yes to accept the agreement and proceed.

i) Next a window will show the completion status as the files are copied. When the copying is complete click on the Finish button to finish the installation.

j) An Aura Server icon will be installed on the desktop.

Installation of Database Script Updates

NOTE

Skip this step if re-installing MPS/Aura Release 2 at a later date or if the files have already been installed from MPS/Aqua Release 6.3.

Copy all of the files in the Database folder to d:\pm1_db\scripts, overwriting any existing files.

Attachment D - Special Operating Instructions

This attachment contains new special operating instructions for MPS/Aura Release 2.0. The information presented in this attachment has been checked for accuracy by the independent test team.

A User's Guide is being updated to include the information presented in this section. The User's Guide will be available from the ETS home page at http://esdis-it.gsfc.nasa.gov/ ETS/ets.html.

PDB Ingest Enhancements

In response to DR ETS0437, two changes have been made to the PM-1 SQL*Plus scripts that ingest the Project Data Base (PDB) into Oracle.

1. It is no longer necessary to copy and rename the PDB flat files into the *pdb_data* folder. A Java program has been provided which performs this function automatically. Simply download the PDB flat files from the Toronto server into a convenient folder and invoke the LOAD_NEXT_PDB batch file. The batch file will execute the Java program, which in turn will prompt for the path to the source folder, then copy and rename the PDB files. There are two restrictions in the use of this program:

   - The source folder chosen should contain only the PDB flat files. Extraneous files would be copied to the *pdb_data* folder along with the PDB flat files.
   - The Java program expects the PDB flat file names to be of the form *tlm_packet_120000.pdb*. All characters from and including the last underscore up to the filename extension (.pdb) are removed.

2. A Java program has been provided that will repair the problem with ingesting the cmd_verify PDB flat file. The program locates records that have only a NULL in the last field. It adds a space to each record found. This program is also executed automatically when the LOAD_NEXT_PDB batch file is invoked.

The *BUILDPM1* batch file has also been updated to execute these Java programs.

TES Segmented Telemetry Packets

In Release 2.0 the MPS/Aura simulator emulates TES Segmented Telemetry by means of a capability similar to dump packets. A 520 byte buffer, FPCResponseData, has been created. This buffer is large enough to contain the largest known TES telemetry data area. A scenario script would be used to trigger the telemetry dump in response to a command being received. A boilerplate scenario script, *TES Boilerplate.txt*, is being provided with the software delivery. Part of this script is shown below. To affect a dump of TES Special Cooler telemetry, the user would preset the FPCResponseData buffer to

the desired data pattern via setBuffer directives in this script, or another one. The scenario script name would have to be entered into the Command-Scenario.txt file, along with the appropriate command mnemonic. When the command is received, it triggers execution of the script, which in turn disables APID 1690, enables APID 1717, waits an appropriate amount of time for APID 1717 telemetry transmission to finish, then finally disables APID 1717 and re-enables APID 1690.

Partial contents of *TES Boilerplate.txt*:

set FPCResponseLength 520
set FPC_ID 2
setBuffer FPCResponseData 0 byte 99 98 97 96 95 94 93 92 91 90 89 88 87 86 85 84 83
setBuffer FPCResponseData 20 byte 79 78 77 76 75 74 73 72 71 70 69 68 67 66 65 64 63
.
.
.
set TlmPacket1690Enabled 0
set TlmPacket1717Enabled 1
sleep 20000
set TlmPacket1717Enabled 0
set TlmPacket1690Enabled 1

Conditional Scenario Script Processing

The capability to include WHILE loops and IF-THEN-ELSE conditional execution within scenario script files is included in this release. Complete usage directions, with examples, may be found in section SN-2.3.1 of the User's Guide. Additional examples are given at the end of this section.

Points to note:

- A WHILE loop must always contain a sleep directive. Failure to do so will cause the loop to execute at full CPU speed, robbing all other functions of processing time.

- The accumulated total of all while loop iterations in scenario scripts running at any given time cannot exceed 1000. Every WHILE loop should include a maximum loop count as part of its condition. Failure to limit the number of consecutive loop iterations can cause the MPS to crash. A WHILE "FOREVER" loop can be implemented by controlling the loop count and having the scenario called again.

- While a conditional expression may be up to 260 characters long, the scenario module currently inserts spaces into the line as it is parsed. It is suggested that users limit conditional expressions to 150 characters.

- Comments and blank line may not contain module specification syntax characters. See the examples below.

- The module specification syntax must contain a space after the #n. Example:

  #2set <mmemonic> 1   WRONG

  #2 set <mnemonic> 1   RIGHT

- Once specified, a module specification affects every line of the script that is executed until an new module specification is issued. Read the comment in the following example.

  ```
  #1 set M1 255
  while (M1 > 0)
      sleep 500
      set M1 -= 2
      ; After the next statement is executed, every directive will be sent to module #2

      #2 set XYZ 5
  endwhile
  ```

- For debugging purposes, all lines of a scenario script may be echoed to the event log. To do so, send the directive

  set Scenario<number>DirMsg 1

  to the Scenario module. Leave "<number>" NULL if the script is executing in the first GUI controlled scenario slot. To set debugging on for the 2nd, 3rd, 4th, and 5th GUI controlled scenario slots, put "1", "2", "3", or "4" into the <number> field.

Examples of Conditional Scenario Scripts

```
; Example 1:
; Conditional Scenario
;
; Multiple IFs test
;
if (CDH_SS_ISASUUSYNC == 0)
   SET CDH_SS_ISASUUSYNC 1

else IF (CDH_SS_PCASULSYNC == 1)
   set CDH_SS_PCASULSYNC 0

ELSE IF (CDH_VA_ISCREF2CALRP == 0)
   set CDH_VA_ISCREF2CALRP 2
endif




;Example 2:
; Conditional Scenario
;
; Loop test
;
set CDH_SS_ISASUUSYNC 0
sleep 1000
while (CDH_SS_ISASUUSYNC < 5)

   set CDH_SS_ISASUUSYNC += 1
   sleep 1000

   if (MOD_CR_SR_GRAT_CH_B == 0)
       SET MOD_CR_SR_GRAT_CH_B 1
   else
       set MOD_CR_SR_GRAT_CH_B 0
   endif

endwhile
; end
```

```
;Example 3:
; Multimodule (EOSGS & SCAura) Conditional Scenario
;
; Loop test
;
#1 set CDH_SS_ISASUUSYNC 0
set CDH_SS_ISASUUSYNC += 1
;
#2 get GSCMDExpectedSCID
;
#1 sleep 1000
while (CDH_SS_ISASUUSYNC < 5)
        set CDH_SS_ISASUUSYNC += 1
        sleep 1000
endwhile
; end
```

```
; Example 4
; While forever workaround
; Construct while loops so that the total never exceeds 1000 iterations.
; If the script must loop for more than that, exit the loop after 1000 iterations
; and use the start scenario directive to call the scenario file again.
;
while (I < 1000)
   set <mnemonic> += 1
   set I += 1
   sleep 500
endwhile
start scenario <my_own_name>
```

CLCW Transmit Rate

The transmit rate of CLCW packets in IP mode may now be changed to match the rate of the telemetry being generated. The container item name is CLCWInterval. It contains the interval between successive transmissions in milliseconds. The following table gives the nominal settings. These settings have no effect when in Serial mode.

H/K Packet Rate	CLCWInterval Setting
16 K (16384)	125
4 K (4096)	500
1 K (1024)	2000

Telemetry Data Value Validation

In response to DR ETS0363, telemetry parameter data values entered by the operator are now checked to ensure that the value will fit into the packet space. Telemetry parameters are treated as unsigned integers for purposes of this validation. If the value entered will not fit, a warning message is sent to the event log and the value is set to zero. Values entered as Engineering Units are converted to the corresponding raw count before the test is performed.

The following is an example of the event message. The user attempted to set parameter CDH_SS_ISASUUSYNC to 266. The value that the user entered is repeated in decimal and in hex, along with the parameter packet size in bits.

SCAura#3: CDH_SS_ISASUUSYNC, TLM#18416: Value 266 (0x10a) too large for 8 bits, set to 0

APID Status Display

A display has been added which shows the current enabled status and transmit rate for all APIDs defined in the Project Data Base. To access this display, click on the *Telemetry* button of the SCAura Control menu. Click on the *APID Status* option from the resulting pull-down menu. The Pause button must be clicked on to maintain the display at the bottom of its list, since it updates every three seconds. More information may be found in Section SCAura-5.2.1.1.9 of the MPS/Aura Release 2.0 User's Guide.

Miscellaneous

- In response to user requests to make the initial setup easier, boilerplate Projects for Serial mode transmission and playback file creation are being delivered with the software.

- New warning and informational messages have been added to the database load logic, and ToolTips have been added to certain GUI screens.

- The directive entry line space has been increased so that 60 characters are displayed. The directive entry line length remains at 99 characters.

- The browse window used to locate scenario files has been fixed so (1) more than 1400 characters of filenames will be displayed, and (2) white space in folder and file names is allowed.

Playback File Creation

Improvements in the ability to generate files of Channel Access Data Units (CADUs) for serial replay are described in this section. The inclusion of the vcProcessor module allows the user to create clean playback files by throwing away fill CADUs.

IMPORTANT

When creating a file of telemetry CADUs for later replay, ensure that the MPS connection to the Matrix Switch is disabled to prevent inadvertent transmission of data.

The procedure for building a file of CADUs is as follows:

1. Build a project with the SCAURA module connected to a Serial Output module, and to vcProcessor and Log modules, as shown in the figure on the second page following. A project with the name of *Playback Creation* is being delivered which automates most of this. The following steps that are asterisked (*) must be completed whether or not a saved project is used.

2. (*) After loading the database, perform the extra SCAURA configuration step of setting the simulation mode to serial. To do this, click on the **Select Simulation Mode** option of the **SCAURA Configuration Menu**, and select **Serial Mode** in the resulting window. See Paragraph SCAURA-5.1 in the User's Guide for more information.

3. Configure the Serial Output module frequency to 16383. See the Serial Output section of the User's Guide. No other setup is necessary.

4. (*) Configure the vcProcessor module as follows. Check the RS box for VC 1. Click the Set Outputs button. In the Channel Selection screen that pops up, check VC 1 for Channel 1. Apply and Close the Channel Selection screen, then Apply and Close the Configuration screen.

5. (*) Configure the Log Module by entering a file name. You may also deselect the **Log With Header** option and the **Variable Length Output** option. If you deselect the **Variable Length Output** option you must ensure that the Packet size is set to 256 bytes.

When replaying the file created, the TxFile module must be set to a complementary configuration. It is suggested that Notepad be used to create a text file containing setup and descriptive information for the log file being created. Give it the same name as the log file, with an extension of .txt, and save it in the same folder as the log file.

---

6. The Virtual Channel Identification (VCID) and Replay fields of the VCDU header must be set to indicate that playback VCDUs are being generated. After running the project, and before starting telemetry transmission, enter the following two set directives to the SCAURA module:

   set vcdu2vcid 1
   set vcdu2replay 1

7. (*) Run the usual telemetry startup scenario file, any special telemetry value setup scenario files desired, then start telemetry transmission by clicking on the **SRL** button of the SCAURA Main Display.

8. At 16 Kbit the file will take 32 times longer to create than to play back at 524 Kbit. For example a four minute playback file will take 128 minutes to create.

9. After the file creation is complete, reset the two VCDU header parameters (see Step 6) to the following values:
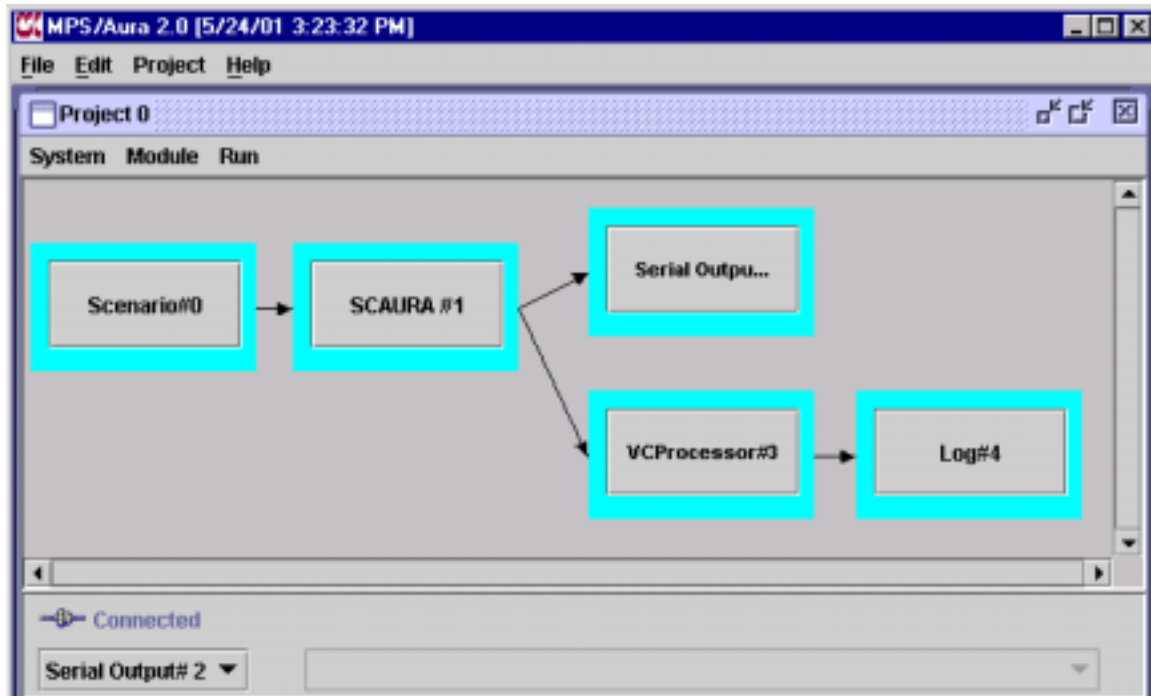
   set vcdu2vcid 2
   set vcdu2replay 0

Figure 1: Example of a Project for Playback File Creation

Attachment E – Resolved Discrepancy Reports

The following Discrepancy Reports (DRs) and Change Requests (CRs) have been closed by and are being delivered with MPS/Aura Release 2.0. Although four of the DRs were written against the MPS/PM-1 simulator they apply to the MPS/Aura simulator as well. The DRs/CRs are listed in the table below, which provides the DR/CR Number, Status, Severity, and a short description. A full description of each DR/CR follows the summary table. Complete information on all DRs/CRs may be accessed via the Internet at address http://iree.gsfc.nasa.gov/ddts/.

Summary of Closed Discrepancy Reports

Critical (Severity 1)	Urgent (Severity 2)	Routine (Severity 3)	Change Requests	Total
0	1	4	1	6

Status Definitions

N – New	A – Assigned Analysis	R – Analysis Entered
V – Assigned Verification	T – Tested	C – Closed
W – Withdrawn	P – Postponed	X – Duplicate

ETS No.	SMO No.	Type	Severity	Version Fixed In	Description
ETS0343	SMOdr05706	DR	3	2.0	Removing Links
ETS0363	SMOdr05924	DR	3	2.0	Telemetry data values entered by operator are not validated.
ETS0406	SMOdr07803	CR	3	2.0	MPS Scenario Execution from Command Subfields
ETS0436	SMOdr09840	DR	3	2.0	Nand and Nor functions don't work
ETS0437	SMOdr10213	DR	3	2.0	Ingest of cmd_verify PDB flat file fails
ETS0441	SMOdr10607	DR	2	2.0	Mis-spelled tlm mnemonic with arithmetic fcn causes server crash.

```
DR:    SMOdr05706 (ETS0343)    Related NCR:        Submitted: 991103
Status: ASSIGNED-ANALYSIS      Class: ETS         Asgnd-Analysis: 991109


  Title: removing links
SUBMITTAL INFORMATION                      ANALYSIS INFORMATION
 Project:        ETS                        Assignee1/Org: Ernest Quintin
 Rel/Ver:        2.1                        Phone: 301-805-3649
 Subsystem:      MPS-PM/Aqua                Email: equintin@csc.com
 Test Phase:     unit test                  Assignee2/Org:
 Severity:       3                          Phone:
 Date found:     991029                     Email:
 Location:       Denver                     Date due (Sev=1,2):
 Submitter:      Vince Ruland
 Organization:   ETS
 Phone number:   720-895-4068
 Email:          vhruland@west.raytheon.com

************ Problem (Added 991103 by vruland) ***********
There should be an easier way to remove an incorrect link or a link made in error
rather than going into the design mode to delete it.  After the link is deleted, the
create links mode has to be re-entered in order to continue creating links.


************ Admin Comments (Added 991109 by eshurie) ***********
Per DRB meeting (11/8/99), the developers stated that the changes
needed for this enhancement will be very involved.
```

```
DR:  SMOdr05924 (ETS0363)     Related NCR:            Submitted: 991217
Status: ASSIGNED-ANALYSIS     Class: ETS              Asgnd-Analysis: 000117


  Title: Telemetry data values entered by operator are not validated
SUBMITTAL INFORMATION                      ANALYSIS INFORMATION
 Project:        ETS                        Assignee1/Org: Ernest Quintin
 Rel/Ver:        3.0                        Phone: 301-805-3649
 Subsystem:      MPS-PM/Aqua                Email: equintin@csc.com
 Test Phase:     acceptance test           Assignee2/Org:
 Severity:       3                          Phone:
 Date found:     991217                     Email:
 Location:       GSFC                       Date due (Sev=1,2):
 Submitter:      Ernest Quintin
 Organization:   ETS Dev Group
 Phone number:   301-805-3649
 Email:          equintin@csc.com
```

************ Problem (Added 991217 by equintin) ***********
No validation is being performed on the values entered in the
Set Directive Window. If the value entered exceeds the number
of bits specified for the telemetry point, high order bits are
truncated when the packet is built. If you enter a hex value,
the telemetry point gets set to zero. Binary values are
interpreted as decimal. In addition, the system does not notify
the user concerning the invalid entry

************ Admin Comment (Modified 010402 by ckolb) ***********
3/30/2001 - Per DRB, this problem will be fixed in Aura Release 2.

01/05/01 DRB: Per developer, the fix for this will be included
in Release 6.2, scheduled for January 26 delivery.

```
DR:  SMOdr07803 (ETS0406)      Related NCR:              Submitted: 000821
Status: ASSIGNED-ANALYSIS      Class: ETS               Asgnd-Analysis: 000929


  Title: MPS Scenario Execution from Command Subfields
SUBMITTAL INFORMATION                      ANALYSIS INFORMATION
 Project:        ETS                       Assignee1/Org: Ernest Quintin
 Rel/Ver:        6.0                       Phone: 301-805-3649
 Subsystem:      MPS-PM/Aqua               Email: equintin@csc.com
 Test Phase:     unit test                 Assignee2/Org:
 Severity:       3                         Phone:
 Date found:     000804                    Email:
 Location:       GSFC                      Date due (Sev=1,2):
 Submitter:      Ed Weidner
 Organization:   Other
 Phone number:   (301)867-0023
 Email:          eweidner@qssmeds.com


************ Configuration (Added 000821 by eweidner) ***********
MPS version 6.0 Beta



************ Problem (Modified 000821 by eshurie) ***********
The MPS does not currently spawn scenarios from command subfields
(via command scenario file), only from the prime command mnemonic.
Since many commands require subfields to designate the prime
action (such as CERES mode commands for example), this greatly
limits the fidelity of the tool.

Note:  This would also require the constraint of only allowing a
command to be listed once in the command scenario file to be
adjusted to either allow multiple times or recognize subfields as
well.



************ Admin Comment (Modified 010402 by ckolb) ***********
3/30/2001 - Per DRB, this problem has been partially fixed in Aura
Release 1, and the final fix will be in Aura Release 2.

9/29/00:  The DRB decided that this is a Change Request and would
be a new requirement.  As such, it is being moved to Analysis -
an estimate of the effort it would require will be made when time
allows.

8/21/00:  Per Carolyn Dent, this DR - Spcrft_A0104 - is really an
ETS problem and will now be moved from Spacecraft in the drtt to
the ETS class in the drtt.  (The spcrft number will no longer exist)
Email dated 8/21/00 follows:

  This appears to be an MPS not a IVVF problem.  The problem
  should be moved to the ETS with the Element MPS-Aqua or MPS.

  Thanks, Carolyn
```

```
DR:  SMOdr09840 (ETS0436)     Related NCR:           Submitted: 010315
Status: ASSIGNED-ANALYSIS     Class: ETS             Asgnd-Analysis: 010402


  Title: Nand and Nor functions don't work
SUBMITTAL INFORMATION                    ANALYSIS INFORMATION
 Project:        ETS                     Assignee1/Org: Ernest Quintin
 Rel/Ver:        1.0                     Phone: 301-805-3649
 Subsystem:      Aura                    Email: equintin@csc.com
 Test Phase:     system I&T              Assignee2/Org:
 Severity:       3                       Phone:
 Date found:     010315                  Email:
 Location:       GSFC                    Date due (Sev=1,2):
 Submitter:      Ernest Quintin
 Organization:   ETS Dev Group
 Phone number:   301-805-3649
 Email:          equintin@csc.com

************ Problem (Added 010315 by equintin) ***********
Please describe the problem you are experiencing below, including
what you did, what you expected to happen, and what actually happened:

The Nor function always returns zero, no matter what input it
is given.

The Nand function always returns an error message indicating that it
cannot find the mnemonic, but it appends the letter n to the mnemonic.
Ex:  mod_cr_sr_grat_ch_bn  not found


************ Admin Comments (Added 010402 by ckolb) ***********
3/30/2001 - Per DRB, this problem will be fixed in Aura Release 2.
```

```
DR:  SMOdr10213 (ETS0437)      Related NCR:              Submitted: 010413
Status: ASSIGNED-ANALYSIS      Class: ETS               Asgnd-Analysis: 010427


  Title: Ingest of cmd_verify PDB flat file fails
SUBMITTAL INFORMATION                    ANALYSIS INFORMATION
 Project:        ETS                      Assignee1/Org: Ernest Quintin
 Rel/Ver:        1.0                      Phone: 301-805-3649
 Subsystem:      Aura                     Email: equintin@csc.com
 Test Phase:     in-field use             Assignee2/Org:
 Severity:       3                        Phone:
 Date found:     010330                   Email:
 Location:       GSFC                     Date due (Sev=1,2):
 Submitter:      Ernest Quintin
 Organization:   ETS Dev Group
 Phone number:   301-805-3649
 Email:          equintin@csc.com
```

************ Problem (Modified 010427 by eshurie) ***********
The ingest of the cmd_verify PDB flat file into Oracle fails part
way through. The SQL*Loader script errors out after encountering
51 records with a NULL in the final column. Additionally, these
records are discarded when they should not be.

NOTE: During 4/27/01 DRB, Ernest pointed out that this DR affects
Aqua as well as Aura.

```
DR:  SMOdr10607 (ETS0441)     Related NCR:           Submitted: 010508
Status: NEW                    Class: ETS
```

```
  Title: Mis-spelled tlm mnemonic with arithmetic fcn causes server crash
SUBMITTAL INFORMATION
 Project:        ETS
 Rel/Ver:        6.3
 Subsystem:      MPS-PM/Aqua
 Test Phase:     dev informal integ
 Severity:       2
 Date found:     010504
 Location:       GSFC
 Submitter:      Ernest Quintin
 Organization:   ETS Dev Group
 Phone number:   301-805-3649
 Email:          equintin@csc.com
```

```
************ Problem (Added 010508 by equintin) ***********
Please describe the problem you are experiencing below, including
what you did, what you expected to happen, and what actually happened:

Entering a set directive to change a telemetry point via one of
the arithmetic functions causes a server crash if the mnemonic
is mis-spelled.

Example:  set CDH_SS_CCASULSYNC  += 2

The mnemonic should be CDH_SS_PCASULSYNC.
```

Attachment F – Unresolved Discrepancy Reports

All of the Discrepancy Reports (DRs) and Change Requests (CRs) listed in the following table were written against the MPS/Aqua simulator. They are being carried forward because the software has been ported to MPS/Aura from that simulator. The table includes the DR/CR Number, Status, Severity, and a short description. A full description of each DR/CR follows the summary table. Complete information on all DRs/CRs may be accessed via the Internet at http://iree.gsfc.nasa.gov/ddts/.

Summary of Open Discrepancy Reports

Critical (Severity 1)	Urgent (Severity 2)	Routine (Severity 3)	Change Request (CR)	Total
0	0	3	0	3

Status Definitions

N – New	A - Assigned Analysis	R - Analysis Entered
V - Assigned Verification	T – Tested	C – Closed
W – Withdrawn	P – Postponed	X – Duplicate

ETS #	SMO No.	Type	Severity	Status	Description
ETS0392	SMOdr06633	DR	3	A	Loading database
ETS0423	SMOdr08499	DR	3	A	MPS-1 Crash During MODIS IOE-1
ETS0438	SMOdr10435	DR	3	N	MPS initial values cause out-of-limit alarms at EMOS.

```
DR:   SMOdr06633 (ETS0392)    Related NCR:       Submitted: 000404
Status: ASSIGNED-ANALYSIS    Class: ETS       Asgnd-Analysis: 000512


  Title: Loading database
SUBMITTAL INFORMATION                      ANALYSIS INFORMATION
 Project:        ETS                        Assignee1/Org: Ernest Quintin
 Rel/Ver:        4.0                        Phone: 301-805-3649
 Subsystem:      MPS-PM/Aqua                Email: equintin@csc.com
 Test Phase:     unit test                  Assignee2/Org:
 Severity:       3                          Phone:
 Date found:     000331                     Email:
 Location:       Denver                     Date due (Sev=1,2):
 Submitter:      Vince Ruland
 Organization:   EMOS
 Phone number:   720-895-4068
 Email:          vhruland@west.raytheon.com
```

************ Problem (Added 000404 by vruland) ***********
Please describe the problem you are experiencing below, including
what you did, what you expected to happen, and what actually happened:

A user should be able to load a new version of the database into
the SC module without having to cycle the entire project first.

Whenever the apply button is clicked for the Load Database on a
SC module, the previous instance of the database should be purged
before the module loads the new instance.

************ Admin Comment (Modified 000515 by eshurie) ***********
At 5/12/00 DRB, the developer got more information on this
request, and will write a memo assessing the difficulty of the
different programming options.

Per DRB on 4/14/00, this DR is on HOLD pending further
contemplation of it by Denver. (Would not be able to be
done within a week.)

```
DR:   SMOdr08499 (ETS0423)    Related NCR:              Submitted: 001113
Status: ASSIGNED-ANALYSIS     Class: ETS               Asgnd-Analysis: 001208


  Title: MPS-1 Crash During MODIS IOE-1
SUBMITTAL INFORMATION                    ANALYSIS INFORMATION
 Project:          ETS                    Assignee1/Org: Ernest Quintin
 DR Type:          Problem                Phone: 301-805-3649
 Rel/Ver:          6.0                    Email: equintin@csc.com
 Subsystem:        MPS-PM/Aqua            Assignee2/Org:
 Module:           Simulator              Phone:
 Affected-Requirement:                    Email:
 Test Phase:       unit test              Date due (Sev=1,2):
 Severity:         3
 Date found:       001107
 Location:         GSFC
 Submitter:        MODIS IOT
 Organization:     Other
 Phone number:     301-614-5025
 Email:            modiot@mcst.gsfc.nasa.gov
```

************ Configuration (Added 001113 by MODIOT) ***********
Please describe the current system configuration:

The MPS-1 (Located in the ISR) was brought up on port A4 to be used by the
SIM-S configuration. The predefined project for server/client version 6.0 was
loaded and database version 000822-005 was ingested. The simulator was
configured for nominal 16kbps telemetry. Several scenario files were run via
the Command-Scenario capability of the simulator to mimic the functionality
of MODIS.

************ Problem (Added 001113 by MODIOT) ***********
Please describe the problem you are experiencing below, including
what you did, what you expected to happen, and what actually happened:

The MPS-1 ran smoothly for the first five hours of the test. There were no
indications of any problems with the simulator.

Abruptly, the server component of the software crashed, leaving the client
component disconnected and "lost". The "Dr. Watson" monitoring software
caught the crash and produced the standard Windows NT illegal exception
violation report. At the time of the crash, the only activity the simulator
was performing was standard telemetry broadcasting (no scenario files were
executing, and no commands were being interpreted).

Attempts to bring up another instance of the server component for the client
component to reconnect to failed to succeed. The client component would not
reconnect. The client component had to be exited, and a new instance started
to successfully connect to the new server component.

The result of recycling the client component was the complete loss of the
state of the simulator at the time of the crash. The client component will

reset all telemetry to a default value when it is started, thereby negating all activities that had been previously performed during the IOE. It took a significant amount of time to restore the simulator's state to what it was prior to the server component crash.

The cause for this software crash was not obvious, and is presently still a mystery. After restarting the server component, it operated without incident for the remainder of the MODIS IOE-1.

[In the past, it has been noted that the MPS server and client components appear to have a memory leak. When the software exits, it does not release the memory allocated to it, etc. Perhaps the crash was caused by over five hours of slow memory leaks. This would explain why the incident did not occur again: the test finished in less than 3 hours after the server component was restarted.]

DR: SMOdr010435 (ETS0438)   Related NCR: Submitted: 010430
Status: NEW   Class: ETS

  Title: MPS initial values cause out-of-limit alarms at EMOS
SUBMITTAL INFORMATION
 Project: ETS
 Rel/Ver: 6.2
 Subsystem: MPS-PM/Aqua
 Test Phase: in-field use
 Severity: 3
 Date found: 010427
 Location: Denver
 Submitter: Ernest Quintin
 Organization: ETS Dev Group
 Phone number: 301-805-3649
 Email: equintin@csc.com

************ Problem (Added 010430 by equintin) ***********
Please describe the problem you are experiencing below, including
what you did, what you expected to happen, and what actually happened:

MPS presets all telemetry points to a midpoint of possible data
values. This causes numerous out-of-limit alarms at EMOS when
MPS telemetry data is initialized.

Reported by Vince Ruland.

Attachment G: Requirements Matrix

The following Level 4 Requirements list has been adapted from the Aura Level 4 Requirements list delivered with Release 1.0. Many of the capabilities resulting from these requirements were ported from the Aqua simulator rather than representing work done to produce the Aura simulator. Those requirements that were satisfied in Release 1.0 and are new to the Aura simulator are marked with an asterisk (*) in the Release column.

The Date of Insert/Update column may be used together with the Comments column to identify changed requirements and provide easy traceability. When a requirement is changed, a new entry, without a requirement number, is added just below the existing entry. Rather than being deleted, the original entry will be marked as being superceded. Original entries may be removed in later versions of the document.

Date of this update: 5/17/2001
General changes: Removed the Comments "NEW", "REWORDED", and "RENUMBERED" from certain requirements.
        Removed those requirements which had previously been marked "to be deleted".

MPS/Aura Requirement	Release	MPS/Aura Requirement Description	Date of Insert/Update	Comments
CMD-01	1	The MPS/Aura simulator shall be capable of receiving command data as UDP command blocks.		
CMD-02	1	The MPS/Aura simulator shall be capable of configuring command receipt processing in IP mode.		
CMD-02.01	1	The MPS/Aura simulator shall be capable of configuring IP mode command receipt to UDP MULTICAST mode.		
CMD-02.02	1	The MPS/Aura simulator shall be capable of configuring IP mode command receipt to any valid UDP MULTICAST IP address.		
CMD-02.03	1	The MPS/Aura simulator shall be capable of configuring IP mode command receipt to any valid UDP MULTICAST Port number.		
CMD-02.04	1	The MPS/Aura simulator shall be capable of configuring IP mode command receipt to any block length between one and 6000 bytes.		
CMD-03	1	The MPS/Aura simulator shall accept operator directives that enable or disable the following elements of the command validation process: Codeblock BCH Parity Validation, Transfer Frame Header Validation, FARM Protocol Validation, and User Command Packet Header Validation.		
CMD-03.01	1	When the Codeblock BCH parity validation element is enabled, the command subsystem will verify for each codeblock of each received CLTU that the BCH parity byte matches a computed value and that the spare bit is equal to zero. If any codeblock of a CLTU fails validation, an event message will be generated and that entire CLTU will be discarded. When this element is disabled, the parity byte will be assumed to be valid.		

MPS/Aura Requirement	Release	MPS/Aura Requirement Description	Date of Insert/Update	Comments
CMD-03.02	1*	When the Transfer Frame Header validation element is enabled, the command subsystem will verify that all of the fields of the Transfer Frame header, except the sequence number, match expected values and ranges as defined in the ICD. If the Transfer Frame Header validation fails, an event message will be generated and the entire Transfer Frame will be discarded. If applicable, the CLCW corresponding to that Transfer Frame's virtual channel will be updated with error information. When this element is disabled, the Transfer Frame header values will be assumed to be valid.	3/7/2001	Spacecraft ID was changed to 0xCC; otherwise no change from Aqua.
CMD-03.03	1	When the FARM validation element is enabled, the command subsystem will verify that the Transfer Frame sequence number is valid as expected for FARM-1 protocol as defined in the ICD. If the FARM validation fails, an event message will be generated and the entire Transfer Frame will be discarded. If applicable, the CLCW corresponding to that Transfer Frame's virtual channel will be updated with error information. When this element is disabled, the Transfer Frame sequence number will be assumed to be valid.		
CMD-03.04	1	When the Command Packet Header validation element is enabled, the command subsystem will verify that the Command Packet Header fields contain valid values as defined in the ICD. If the Command Packet Header validation fails, an event message will be generated and the Command Packet will be discarded. This requirement is applicable to the spacecraft command packet format and the instrument command packet format. When this element is disabled, the Command Packet Header is assumed to be valid.		
CMD-04	1	The MPS/Aura simulator shall accept operator directives to change all fields of the spacecraft and instrument CLCWs.		
CMD-05	1	The MPS/Aura simulator shall simulate spacecraft command acceptance according to the COP-1 protocol.		
CMD-05.01	1	The MPS/Aura simulator shall perform Type AD spacecraft command acceptance checks in accordance with the FARM-1 protocol if FARM-1 protocol checking is enabled.		
CMD-05.01.1	1	The MPS/Aura simulator shall reject Type AD spacecraft commands and post a command rejected event message if the Lockout bit is set in the spacecraft CLCW.		
CMD-05.01.2	1	The MPS/Aura simulator shall reject Type AD spacecraft commands, post a command rejected message, and set the Lockout bit in the spacecraft CLCW if (1) the Frame Sequence Count in the Transfer Frame header is more than 90 counts greater than or more than 90 counts less than (modulo 256) the Report Value field of the spacecraft CLCW and (2) FARM-1 protocol checking is enabled.		

Attachment G: Requirements Matrix

MPS/Aura Requirement	Release	MPS/Aura Requirement Description	Date of Insert/Update	Comments
CMD-05.01.3	1	The MPS/Aura simulator shall reject Type AD spacecraft commands, post a command rejected message, and set the Retransmit bit in the spacecraft CLCW if (1) the Frame Sequence Count in the Transfer Frame header is between one and 90 counts greater than (modulo 256) the contents of the Report Value field of the spacecraft CLCW <u>and</u> (2) FARM-1 protocol checking is enabled.		
CMD-05.01.4	1	The MPS/Aura simulator shall reject Type AD spacecraft commands and post a command rejected message if (1) the Frame Sequence Count in the Transfer Frame header is between one and 90 counts less than (modulo 256) the contents of the Report Value field of the spacecraft CLCW <u>and</u> (2) FARM-1 protocol checking is enabled.		
CMD-05.01.5	1	The MPS/Aura simulator shall clear the spacecraft CLCW Lockout bit upon receipt of an UNLOCK Control Command (Type BC) containing the spacecraft VCID.		
CMD-05.01.6	1	The MPS/Aura simulator shall set the spacecraft CLCW Report Value field to the data value contained within the third byte of a SET V(R) Control Command (Type BC) containing the spacecraft VCID.		
CMD-05.01.7	1	The MPS/Aura simulator shall increment the Report Value field (modulo 256) of the spacecraft CLCW upon receipt of a Type AD spacecraft command whose Frame Sequence Count matches the current spacecraft CLCW Report Value field contents, provided that FARM-1 protocol checking is enabled.		
CMD-05.02	1	The MPS/Aura simulator shall perform Type AD instrument command acceptance checks in accordance with the FARM-1 protocol if FARM-1 protocol checking is enabled.		
CMD-05.02.1	1	The MPS/Aura simulator shall reject Type AD instrument commands and post a command rejected event message if the Lockout bit is set in the instrument CLCW.		
CMD-05.02.2	1	The MPS/Aura simulator shall reject Type AD instrument commands, post a command rejected message, and set the Lockout bit in the instrument CLCW if (1) the Frame Sequence Count in the Transfer Frame header is more than 90 counts greater than or more than 90 counts less than (modulo 256) the Report Value field of the instrument CLCW <u>and</u> (2) FARM-1 protocol checking is enabled.		
CMD-05.02.3	1	The MPS/Aura simulator shall reject Type AD instrument commands, post a command rejected message, and set the Retransmit bit in the instrument CLCW, if (1) the Frame Sequence Count in the Transfer Frame header is between one and 90 counts greater than (modulo 256) the Report Value field of the instrument CLCW <u>and</u> (2) FARM-1 protocol checking is enabled.		

Attachment G: Requirements Matrix

MPS/Aura Requirement	Release	MPS/Aura Requirement Description	Date of Insert/Update	Comments
CMD-05.02.4	1	The MPS/Aura simulator shall reject Type AD instrument commands and post a command rejected message if (1) the Frame Sequence Count in the Transfer Frame header is between one and 90 counts less than (modulo 256) the Report Value field of the instrument CLCW <u>and</u> (2) FARM-1 protocol checking is enabled.		
CMD-05.02.5	1	The MPS/Aura simulator shall clear the instrument CLCW Lockout bit upon receipt of an UNLOCK Control Command (Type BC) containing the instrument VCID.		
CMD-05.02.6	1	The MPS/Aura simulator shall set the instrument CLCW Report Value field to the data value contained within the third byte of a SET V(R) Control Command (Type BC) containing the instrument VCID.		
CMD-05.02.7	1	The MPS/Aura simulator shall increment the Report Value field (modulo 256) of the instrument CLCW upon receipt of a Type AD instrument command whose Frame Sequence Count matches the current instrument CLCW Report Value field contents, providing that FARM-1 protocol checking is enabled.		
CMD-06	1	The MPS/Aura simulator shall provide the capability to monitor and display command processing status.		
CMD-07	1	Upon operator request, the MPS/Aura simulator shall store received commands for posttest review subject to specified storage capacities.		
CMD-08	1	The MPS/Aura simulator shall use information from the PDB to perform command identification processing. The Command subsystem shall match command bit patterns received to stored bit patterns to locate command mnemonics in the PDB.		
CMD-09	1	The MPS/Aura simulator shall provide the capability to respond to that subset of spacecraft commands that are defined in the Aura PDB Command Execution Verification (CEV) file. If the PDB CEV file contains end-item verifier telemetry mnemonics associated with the identified command, the associated telemetry point(s) will be set to the corresponding value(s) defined in the CEV file.		
CMD-10	1	The MPS/Aura simulator shall generate a simulator event message whenever a command is received.		
CMD-10.01	1	The MPS/Aura simulator shall generate a simulator event message to display the command mnemonic whenever a valid command is decoded.		
CMD-10.02	1	The MPS/Aura simulator shall generate a simulator event message to display the values of command submnemonics whenever a command containing submnemonics is decoded.		
CMD-10.03	1*	The MPS/Aura simulator shall provide for the storage of command submnemonic values, to be viewable by the operator, for the life of a simulations session.		

Attachment G: Requirements Matrix

MPS/Aura Requirement	Release	MPS/Aura Requirement Description	Date of Insert/Update	Comments
CMD-11	1	The MPS/Aura simulator shall generate a simulator event message whenever a command error is detected		
CMD-11.01	1	The MPS/Aura simulator shall generate a simulator event message indicating the command error detected whenever a command in error is decoded, provided that command validation is enabled.		
CMD-11.02	1	The MPS/Aura simulator shall generate an event message indicating that an unknown command has been received whenever a command cannot be matched to any PDB entry.		
CMD-12	1	The MPS/Aura simulator shall perform verification of selected fields of the Command Data Block (CDB) header of received commands. The fields to be verified shall be Message Type, Source, Destination, spacecraft identifier (SCID), and Sequence Count.		
CMD-12.01	1	The MPS/Aura simulator shall generate event messages reporting inconsistencies in the verifiable fields of the CDB.		
CMD-12.02	1*	The MPS/Aura simulator shall accept and execute operator directives that set expected values for verification of the CDB header.	3/7/2001	Added logic to EOSGS to accept expected Spacecraft ID
CMD-12.03	1	The MPS/Aura simulator shall permit the operator to enable and disable CDB verification.		
CMD-13	1	The MPS/Aura simulator shall receive spacecraft memory and table loads via command blocks and shall store the load data in a load buffer (simulated spacecraft memory.)		
CMD-13.01	1	The MPS/Aura simulator shall perform a validation of the command load data checksum, for those loads that contain a checksum.		
CMD-13.02	1	The MPS/Aura simulator shall permit the operator to inhibit the checksum validation.		
CMD-14	1	The MPS/Aura simulator shall be capable of simulating a spacecraft memory dump of loaded data.		
CMD-14.01	1	The MPS/Aura simulator shall be capable of copying a single memory load from the load buffer to the dump buffer.		
CMD-14.02	1	The MPS/Aura simulator shall permit the operator to inhibit copying the memory load to the dump buffer.		
CMD-15	1	The MPS/Aura simulator shall process commands that request or configure for a spacecraft memory dump.		
CMD-16	1	The MPS/Aura simulator interface with EOC shall comply with the command interface formats and protocols specified in the EDOS to EGS Elements interface document		
CMD-17	1	The MPS/Aura simulator shall be capable of updating multiple command counters in telemetry. The command counters to be updated shall be as agreed upon with Aura project representatives.		

Attachment G: Requirements Matrix

MPS/Aura Requirement	Release	MPS/Aura Requirement Description	Date of Insert/Update	Comments
CMD-18	1	The MPS/Aura simulator shall interpret VCID 0 (spacecraft), VCID 1 (instrument), and VCID 16 and 17 (TIE critical) commands.		
CMD-19	1	The MPS/Aura simulator shall be capable of interpreting multipart commands.		
CMD-20	1	The MPS/Aura simulator shall be capable of logging up to 8 MB of received commands during a testing session.		
CMD-21	1	The MPS/Aura simulator shall be capable of receiving spacecraft commands in a CLTU bitstream through the serial interface at rates from 125 bps to 2 Kbps.		
CMD-22	3	The MPS/Aura simulator shall be capable of receiving spacecraft command packets via a 1553B bus interface.		See also requirement GEN-21.
CMD-22.01	3	The MPS/Aura simulator shall be capable of connecting to multiple Remote Terminal ports when configured to accept commands via a 1553B bus interface.	5/17/2001	NEW.
CMD-22.02	3	The MPS/Aura simulator shall be capable of maintaining a count of commands received from each active Remote Terminal when configured to accept commands via a 1553B bus interface.	5/17/2001	NEW.
CMD-22.03	3	The MPS/Aura simulator shall be capable of bypassing Codeblock BCH Parity Validation, Transfer Frame Header Validation, FARM Protocol Validation, and User Command Packet Header Validation when configured to accept commands via a 1553B bus interface.	5/17/2001	NEW
CMD	TBD	The MPS/Aura simulator shall be capable of a <TBN> simulation of spacecraft Stored Command Processing	3/7/2001	Proposed. TBN.
CMD	TBD	The MPS/Aura simulator shall be capable of receiving instrument memory and table loads via command blocks and shall store the load data in a load buffer (simulated memory.)	3/7/2001	Proposed. Exact capability is TBN.
GEN-01	1	The MPS/Aura simulator shall be Year 2000 compliant		
GEN-02	1	The MPS/Aura simulator shall be capable of maintaining an internally generated time code to a resolution of 125 milliseconds..		
GEN-02.01	1	The MPS/Aura simulator shall be capable of setting GMT and simulated spacecraft time as directed by the operator.		
GEN-03	1	The MPS/Aura simulator shall be capable of executing a scenario script file.		
GEN-03.01	1	The MPS/Aura simulator shall be capable of executing operator directives via a scenario script to update telemetry parameters by mnemonic.		
GEN-03.02	1	The MPS/Aura simulator shall be capable of executing operator directives via a scenario script to retrieve and display the value of any telemetry parameter by mnemonic.		

MPS/Aura Requirement	Release	MPS/Aura Requirement Description	Date of Insert/Update	Comments
GEN-03.03	1	The MPS/Aura simulator shall be capable of executing operator directives via a scenario script to start and stop telemetry transmission.		
GEN-03.04	1	The MPS/Aura simulator shall be capable of executing operator directives via a scenario script to start and stop transmission of CLCW packets.		
GEN-03.05	1	The MPS/Aura simulator shall be capable of executing operator directives via a scenario script to enable and disable all elements of command validation that are under operator control. See "CMD" requirements for those command validation elements that are controllable by the operator.		
GEN-03.06	TBD	The MPS/Aura simulator shall provide a scenario file container name verification capability, the purpose of which is to verify that all telemetry and command mnemonics appearing in scenario scripts exist in the PDB.	5/17/2001	Proposed.
GEN-04	1	The MPS/Aura simulator shall be capable of providing files of received or generated test data on electronic and physical media.		
GEN-05	1	The MPS/Aura simulator shall acknowledge an operator request within 2 seconds of its entry.		
GEN-06	1	The MPS/Aura simulator shall start execution of an operator request within 5 seconds of its entry.		
GEN-08	1	The MPS/Aura simulator shall comply with the set of display guidelines specified in DSTL-92-007, Human-Computer Interface Guidelines, August, 1992.		
GEN-09	1	The MPS/Aura simulator shall comply with security provisions specified in the NASA Automated Information Security Handbook, NHB 2410.9A.		
GEN-10	1	The MPS/Aura simulator shall comply with the NASA Communications (Nascom) Access Protection Policy and Guidelines.		
GEN-11	1	The MPS/Aura simulator shall provide a hard disk drive with sufficient capacity to store the program bootstrap, executable files, and other simulation environment files, such as the Project Data Base and scenario files used during tests, and a TBD percent reserve.		
GEN-12	1	The MPS/Aura simulator shall provide a physical media storage device that can be used to support the exchange of small amounts of information with external systems and for system backups and data logging.		
GEN-13	1	The MPS/Aura simulator shall be portable.		
GEN-14	1	The MPS/Aura simulator shall provide an Ethernet interface that conforms to 10BaseT of the IEEE 802.3 standard.		
GEN-15	1	The MPS/Aura simulator shall interface with the EOC through the Ethernet interface using the Internet Protocol (IP) suite, including TCP/IP and UDP/IP.		

MPS/Aura Requirement	Release	MPS/Aura Requirement Description	Date of Insert/Update	Comments
GEN-16	1	The MPS/Aura simulator shall receive CLTUs in command data blocks from the EOC and output EDUs (packets and CLCWs) to the EOC through the Ethernet interface. All data transfers through this Ethernet interface shall be based on UDP/IP protocol.		
GEN-17	1	The MPS/Aura simulator shall provide a capability to permit modification of the Command End-Item Verifiers file.		
GEN-17.01	1	The MPS/Aura simulator Command End-Item Verifiers modification utility shall permit the addition of a single record to the file of end-item verifiers at each invocation.		
GEN-17.02	1	The MPS/Aura simulator Command End-Item Verifiers modification utility shall permit the modification of a single record of the file of end-item verifiers at each invocation. The Low Limit and State Text fields shall be the only fields that may be modified.		
GEN-17.03	1	The MPS/Aura simulator Command End-Item Verifiers modification utility shall permit the deletion of a single record from the file of end-item verifiers at each invocation.		
GEN-18	1	The MPS/Aura simulator shall provide a file selection browse capability.		
GEN-19	1	The MPS/Aura simulator shall provide a configuration save and restore capability.		
GEN-19.01	1	The MPS/Aura simulator shall be capable of saving module configuration information. The information saved shall consist of the modules that constitute a project, and the module links and link source/destination numbers.		
GEN-19.02	1	The MPS/Aura simulator shall be capable of saving configuration information for IP modules, log modules, and Serial modules.		
GEN-19.03	1	The MPS/Aura simulator shall be capable of saving multiple configurations in separate disk files.		
GEN-19.04	1	The MPS/Aura simulator shall permit the operator to name a disk file in which configuration information shall be saved.		
GEN-19.05	1	The MPS/Aura simulator shall permit the operator to restore configuration information upon initialization.		
GEN-19.06	1	The MPS/Aura simulator shall be capable of restoring configuration information from an existing named disk file.		
GEN-19.07	1	The MPS/Aura simulator shall be capable of displaying the names of the disk files containing configuration information when responding to a restore request during initialization.		
GEN-19.08	TBD	The MPS/Aura simulator shall be capable of displaying the file creation date when responding to a restore request during initialization.		This is a function of the SIMSS baseline. The SIMSS release it will be in has not yet been decided.

Attachment G: Requirements Matrix

MPS/Aura Requirement	Release	MPS/Aura Requirement Description	Date of Insert/Update	Comments
GEN-20	1	The MPS/Aura simulator shall be capable of executing multiple scenario script files simultaneously, up to the limit imposed by CPU and memory capacities.		
GEN-20.01	1	The MPS/Aura simulator shall permit the operator full control of scenario script files that the operator has invoked. The control directives available shall consist of START, STOP, PAUSE, and RESUME.		
GEN-20.02	1	The MPS/Aura simulator shall be capable of starting a scenario script in response to a command received. The operator shall not be able to STOP, PAUSE, or RESUME a scenario script started this way.		
GEN-20.03	1	The MPS/Aura simulator shall be capable of invoking a scenario script from within a scenario script. The operator shall not be able to STOP, PAUSE, or RESUME a scenario script started this way.		
GEN-20.04	1	The MPS/Aura simulator shall be capable of displaying the status of all scenario scripts that were started by the operator. This status shall consist of an indication as to whether the scenario script is running, paused, or finished, a display of the current line number, and a display of the directive currently being executed.		
GEN-20.05	2	Scenario script processing shall be enhanced to include if/then/else or loop execution based on values in container items in the current module.	5/17/2001	SUPERCEDED.
	2	The MPS/Aura simulator shall be capable of executing a scenario script that contains nested if/then/else and while loop constructs.	5/17/2001	
GEN-20.06	2	The MPS/Aura simulator Scenario script processor shall be capable of connecting to up to four modules and sending script directives to the proper module. The operator must provide a module indicator.	5/17/2001	NEW.
GEN-20.06.1	2	The module indicator format in scenario scripts shall be of the form #n, where n is the link number.	5/17/2001	NEW.
GEN-20.06.2	2	When starting execution of any scenario script, the MPS/Aura simulator Scenario script processor shall default the module indicator to link #1. In the absence of a module indicator in a scenario script, the script processor shall send all directives to the module connected to link #1.	5/17/2001	NEW.
GEN-20.06.3	2	When a module indicator is encountered, the MPS/Aura simulator Scenario script processor shall send that and all following directives to the link number indicated, until another module indicator is encountered.	5/17/2001	NEW.
GEN-21	3	The MPS/Aura simulator shall provide a 1553B bus interface for the purpose of receiving command packets and transmitting telemetry packets.		See GUI-17, CMD-22, and TLM-37 requirements for further details.

Attachment G: Requirements Matrix

MPS/Aura Requirement	Release	MPS/Aura Requirement Description	Date of Insert/Update	Comments
GEN-21.01	3	The MPS/Aura simulator shall be capable of emulating up to four Remote Terminals when connected to a 1553B bus.	5/17/2001	NEW.
GEN-21.02	3	The MPS/Aura simulator, when connected to a 1553B bus, shall be capable of maintaining and displaying status counts of the number of commands received and telemetry packets transmitted through each Remote Terminal.	5/17/2001	NEW.
GEN	TBD	The MPS/Aura simulator shall be capable of selecting packets or VCDUs to be logged from a stream of mixed packets or VCDUs based on <TBN> criteria.		Proposed.
GEN	TBD	The MPS/Aura simulator shall provide a limited command generation capability, the purpose of which is to facilitate verification of simulator operational readiness.		
GEN	TBD	The MPS/Aura simulator shall provide a limited telemetry data quality monitoring capability, the purpose of which is to facilitate verification of simulator operational readiness.		
GUI-01	1	The MPS/Aura simulator shall accept and validate all operator directives.		
GUI-01.01	1	The MPS/Aura simulator GUI shall maintain a history list of directives entered by the operator. This history list shall store a maximum of 10 operator directives.		
GUI-01.02	1	The MPS/Aura simulator GUI shall permit the operator to re-execute directives stored in the history list.		
GUI-01.03	1	The MPS/Aura simulator GUI shall permit the operator to edit directives stored in the history list.		
GUI-02	1	The MPS/Aura simulator GUI shall provide the capability to display command packets received.		
GUI-03	1	The MPS/Aura simulator GUI shall provide the capability to display telemetry and CLCW packets transmitted.		
GUI-04	1	The MPS-Aura simulator GUI shall provide the capability to display command and telemetry status.		
GUI-05	1	The MPS/Aura simulator GUI shall provide the capability to display the current receive and transmit network configuration to the operator.		
GUI-07	1	The MPS/Aura simulator shall provide the capability to display the EDOS Service Header appended to transmitted telemetry packets.		
GUI-08	1	The MPS/Aura simulator shall provide the capability to display the Telemetry Packet Header of a selected APID.		
GUI-09	1	The MPS/Aura simulator shall provide the capability to display GMT and Spacecraft times.		

Attachment G: Requirements Matrix

MPS/Aura Requirement	Release	MPS/Aura Requirement Description	Date of Insert/Update	Comments
GUI-10	1	The MPS/Aura simulator shall provide the capability to display the current values of the spacecraft and instrument CLCWs.		
GUI-11	1	The MPS/Aura simulator shall provide the capability to display event messages.		
GUI-11.01	1*	The MPS/Aura simulator shall provide the operator with the capability to suppress display of event messages. The filtering mechanism shall be keyed to the event message color.		
GUI-11.02	1	The MPS/Aura simulator shall log all generated event messages to a disk file.		
GUI-12	1	The MPS/Aura simulator shall provide the capability to display telemetry and CLCW transmit status.		
GUI-13	1	The MPS/Aura simulator shall provide the capability to display command receipt status.		
GUI-14	1	The MPS/Aura simulator shall be capable of updating all displays periodically.		
GUI-15	1	The MPS/Aura simulator shall provide a generic buffer display.		
GUI-16	2	The MPS/Aura simulator shall provide a single display giving the telemetry packet enable status and transmit interval for all APIDs in the PDB.		
GUI-17	3	The MPS/Aura simulator shall provide configuration and status displays for the 1553B bus interface.	5/17/2001	NEW. See CMD-22, GEN-21, and TLM-37 requirements for further details.
GUI-17.01	3	The MPS/Aura simulator 1553B bus interface configuration display shall permit the operator to enter one to four Remote Terminal numbers and associate a range of telemetry APIDs to be transmitted over each.	5/17/2001	NEW.
GUI-17.02	3	The MPS/Aura simulator 1553B bus interface status display shall display the number of command packets received and telemetry packets transmitted for each Remote Terminal connected.	5/17/2001	NEW.
INIT-01	1(3)	The MPS/Aura simulator shall be capable of selecting a desired version of the PDB at operator request during initialization.	5/17/2001	Differences between the Aqua and Aura PDB schema will require a revisit of this requirement when the Aura PDB is finalized.
INIT-01.01	1	During initialization, the MPS/Aura simulator shall provide the operator with the capability to select one version of the Aura PDB from among those available.		
INIT-01.02	1	During initialization, if the operator does not select a version of the Aura PDB, the MPS/Aura simulator will default to the most recent version available.		

Attachment G: Requirements Matrix

MPS/Aura Requirement	Release	MPS/Aura Requirement Description	Date of Insert/Update	Comments
INIT	TBD	Dependence upon Oracle as a database repository shall be removed.	3/7/2001 5/17/2001	NEW. The SIMSS baseline software is being modified to ingest PDB flat files directly during initialization. The completion date of this activity is TBD. TO BE REMOVED. The SIMSS PDB flat file capability will not be ready in time.
MDL-01	TBD	The MPS/Aura simulator shall provide a telemetry parameter modeling capability. The purpose of this capability is to simulate the behavior of a limited set of telemetry parameters.	3/7/2001	The modeling requirements (MDL-n) are Placeholders. The MPS/Aura simulator shall inherit the SIMSS Modeling capability when it achieves sufficient maturity. The completion date of that activity is TBD. The exact modeling requirements are also TBD.
MDL-02		The MPS/Aura simulator shall turn on and off selected modeling under operator control.		
MDL-03		The MPS/Aura simulator shall be capable of changing between static, table, or algorithm models under operator control		
MDL-04		The MPS/Aura simulator shall accept and execute modeling directives that enable or disable selected modeling.		
MDL-05		The MPS/Aura simulator shall accept and execute modeling directives that associate any telemetry parameter with any predefined model.		
MDL-06		The MPS/Aura simulator shall accept and execute modeling directives that change between static, table, or algorithm models.		
MDL-07		The MPS/Aura simulator shall provide the operator with an offline capability to access model functions and coefficients.		
MDL-08		The MPS/Aura simulator shall provide the operator with an offline capability to translate ASCII-formatted files containing static, table, and algorithm modeling information into a binary form readable by The MPS/Aura simulator.		
TLM-01	1	The MPS/Aura simulator shall be capable of switching between IP and serial modes of operation for command receipt and telemetry transmission.		
TLM-02	1	The MPS/Aura simulator shall provide the capability to transmit one stream of telemetry when in IP mode.		
TLM-03	1	The MPS/Aura simulator shall be capable of independently configuring telemetry and CLCW transmit when in IP mode.		

Attachment G: Requirements Matrix

MPS/Aura Requirement	Release	MPS/Aura Requirement Description	Date of Insert/Update	Comments
TLM-03.01	1	The MPS/Aura simulator shall be capable of transmitting packets containing CLCWs independently of telemetry transmission when in IP mode		
TLM-03.02	1	The MPS/Aura simulator shall default the CLCW packet transmission rate to 8 packets per second.		
TLM-03.03	2	The MPS/Aura simulator shall be capable of adjusting the frequency of CLCW packet transmission under operator control. The purpose of this requirement is to permit the CLCW transmission rate to match that of the telemetry transmission.		
TLM-03.04	1	The MPS/Aura simulator shall be capable of independently configuring IP mode telemetry and CLCW transmission to UDP MULTICAST mode when in IP mode.		
TLM-03.05	1	The MPS/Aura simulator shall be capable of independently configuring IP mode telemetry and CLCW transmission to any valid UDP MULTICAST IP address when in IP mode.		
TLM-03.06	1	The MPS/Aura simulator shall be capable of independently configuring IP mode telemetry and CLCW transmission to any valid UDP MULTICAST Port number when in IP mode.		
TLM-03.07	1	The MPS/Aura simulator shall be capable of independently configuring IP mode telemetry and CLCW transmission to any block length between one and 6000 bytes when in IP mode.		
TLM-03.08	1	The MPS/Aura simulator shall be capable of independently configuring IP mode telemetry transmission to variable block length when in IP mode.		
TLM-04	1	The MPS/Aura simulator shall be capable of transmitting one stream of CADUs when in serial mode.		
TLM-04.01	1	When in serial mode, the MPS/Aura simulator shall build S-band I-Channel CADUs as described in the Aura Spacecraft to Ground ICD.		
TLM-04.02	1	When in serial mode, the MPS/Aura simulator shall build and transmit I-Channel Fill CADUs as described in the Aura Spacecraft to Ground ICD when there is not enough telemetry data available to fill a CADU.		
TLM-05	1	The MPS/Aura simulator shall accept and execute operator directives that set the value of any telemetry parameter by mnemonic and by parameter ID.	5/17/2001	REWORDED.
TLM-05.01	1	The MPS/Aura simulator shall permit the operator to update telemetry parameter values in decimal, hex, and octal raw data numbers, and in Engineering Units.	5/17/2001	RENUMBERED from TLM-07.03

Attachment G: Requirements Matrix

MPS/Aura Requirement	Release	MPS/Aura Requirement Description	Date of Insert/Update	Comments
TLM-05.02	1*	The MPS/Aura simulator shall be capable of accepting directives to set telemetry values using simple expressions (addition, subtraction, multiplication, etc.), trigonometric functions (sin, cos, etc.), Boolean expressions, the values or other telemetry parameters, and any combination thereof.	5/17/2001	RENUMBERED from TLM-07.07
TLM-05.03	2	The MPS/Aura simulator shall be capable of limit checking operator-entered telemetry values and informing the operator if a value is too large to fit into the packet space. If a value is too large for the packet space it will be set to zero.	5/17/2001	NEW
TLM-06	1*	The MPS/Aura simulator shall be capable of accessing telemetry parameters by mnemonic and by parameter ID.	5/17/2001	RENUMBERED from TLM-07.06. Parameter ID is also known as Last Received Value (LRV).
TLM-06.01	1	The MPS/Aura simulator shall use the PDB to define raw-data-to-EU and EU-to-raw-data conversions for telemetry parameters.	5/17/2001	RENUMBERED from TLM-07.02
TLM-07	1	The MPS/Aura simulator shall accept and execute operator directives that request the value of any telemetry parameter for display.		
TLM-07.01	1	The MPS/Aura simulator shall be capable of displaying telemetry parameter values in decimal raw counts and in Engineering Units		
TLM-07.02	1	The MPS/Aura simulator shall be capable of displaying multiple telemetry parameter values in a GUI window.		
TLM-07.03	1	The MPS/Aura simulator shall be capable of displaying multiple iterations of a GUI window for display and update of telemetry parameters.		
TLM-08	1	The MPS/Aura simulator shall accept and execute operator directives that request the contents of any telemetry packet.		
TLM-09	1	The MPS/Aura simulator shall accept and execute operator directives that set the value of any location in the Aura-simulated spacecraft memory.	5/17/2001	RENUMBERED from TLM-06
TLM-09.01	1	The MPS/Aura simulator shall accept and execute operator directives that request the value of any location or block of locations in simulated spacecraft memory.		
TLM-10	1	The MPS/Aura simulator shall set initial telemetry parameter values from information extracted from the Aura PDB.		
TLM-11	1	The MPS/Aura simulator shall accept and execute operator directives that result in changes to telemetry packet header values.		
TLM-12	TBD	The MPS/Aura simulator shall accept and execute telemetry directives that control the Aura Solid State Recorder.	5/17/2001	SUPERCEDED.

MPS/Aura Requirement	Release	MPS/Aura Requirement Description	Date of Insert/Update	Comments
	TBD	The MPS/Aura simulator shall be capable of a <TBN> emulation of Solid State Recorder operation.	3/7/2001 5/17/2001	PLACEHOLDER. Exact requirement is TBN. A low fidelity emulation may be achieved through the use of scenario scripts. REWRITTEN.
TLM-13	1	The MPS/Aura simulator shall provide for the storage of housekeeping telemetry to be used as playback data.		
TLM-14	1	The MPS/Aura simulator shall use the information from the Aura PDB to generate and transmit telemetry packets.		
TLM-14.01	1	The MPS/Aura simulator shall be capable of creating CCSDS-format telemetry packets from information contained in the Aura PDB telemetry packet specification file.		
TLM-14.02	1	The MPS/Aura simulator shall provide the capability to generate and transmit telemetry packets with APIDs identical to the Aura spacecraft.		
TLM-14.03	1	The MPS/Aura simulator shall use the APID and secondary key fields of the Aura PDB packet definition file to identify unique packets.		
TLM-14.04	1	The MPS/Aura simulator shall generate a telemetry packet for each unique combination of APID and secondary key.		
TLM-14.05	1	The MPS/Aura simulator shall accept and execute operator directives to set the packet generation rate for any APID defined in the Aura PDB.		
TLM-14.06	1	The MPS/Aura simulator shall populate the telemetry packet primary header fields in the following list in accordance with information obtained from the Aura Spacecraft to Ground ICD and applicable CCSDS documents: Version Number, Type, Secondary Header Flag, APID, Sequence Flag, Sequence Count, and Packet Length.		
TLM-14.07	1	The MPS/Aura simulator shall place the secondary key into the telemetry packet at the offset specified by the Aura PDB telemetry packet specification file and shall use the number of bits specified by that file.		
TLM-14.08	1	The MPS/Aura simulator shall generate a telemetry packet secondary header in accordance with the secondary header type (SC, GIRD, or None for SUROM-TIE packets) implied by the contents of the packet type field of the Aura PDB telemetry packet specification file. For each secondary header type, the contents shall be as described in applicable sections of the Aura Spacecraft to Ground ICD.		
TLM-15	1	The MPS/Aura simulator shall insert simulated spacecraft time in the telemetry packet headers		

MPS/Aura Requirement	Release	MPS/Aura Requirement Description	Date of Insert/Update	Comments
TLM-16	1	The MPS/Aura simulator shall maintain data values for all telemetry parameters defined in the PDB telemetry parameter specification file. These data values shall be available for display to the operator and for inclusion into telemetry packets.		
TLM-16.01	1	The MPS/Aura simulator shall be capable of inserting telemetry point values into packets using information from the Aura PDB telemetry description and telemetry parameter specification files.		
TLM-16.02	1	The MPS/Aura simulator shall use the APID and secondary key fields of the PDB telemetry parameter specification file to determine the correct packet for each telemetry parameter.		
TLM-17	1	The MPS/Aura simulator shall send out telemetry packets at specified intervals of spacecraft time. These specific intervals shall be as defined by the PDB for each APID and secondary key combination and shall be modifiable by the operator.		
TLM-18	1	The MPS/Aura simulator shall be capable of simulating spacecraft memory dumps. The MPS/Aura simulator shall build packets based on the contents of the simulated spacecraft memory.		
TLM-19	1*	The MPS/Aura simulator shall accept and execute directives that start and stop transmission of telemetry data.		
TLM-19.01	1*	The MPS/Aura simulator shall start transmission of telemetry and CLCW packets upon receipt of a start telemetry directive when in IP mode, unless startup of CLCW packet transmission is inhibited. In that case only telemetry packet transmission shall be started.	3/7/2001	SUPERCEDED.
	1*	The MPS/Aura simulator shall start transmission of telemetry and CLCW packets upon receipt of a start telemetry directive when in IP mode, unless startup of CLCW packet transmission is unlinked from telemetry startup. In that case only telemetry packet transmission shall be started.	3/7/2001	
TLM-19.02	1*	The MPS/Aura simulator shall stop transmission of telemetry and CLCW packets upon receipt of a stop telemetry directive when in IP mode.	3/7/2001	SUPERCEDED.
	1*	The MPS/Aura simulator shall stop transmission of telemetry and CLCW packets upon receipt of a stop telemetry directive when in IP mode, unless stopping of CLCW packet transmission is unlinked from telemetry stop. In that case only telemetry packet transmission shall be stopped.	3/7/2001	
TLM-19.03	1*	The MPS/Aura simulator shall be capable of starting and stopping the transmission of CLCW packets independently of telemetry transmission when in IP mode.		

Attachment G: Requirements Matrix

MPS/Aura Requirement	Release	MPS/Aura Requirement Description	Date of Insert/Update	Comments
TLM-19.04	1*	The MPS/Aura simulator shall be capable of setting a flag, under operator control, which, when set, shall inhibit startup of CLCW packet transmission when telemetry transmission is started in IP mode.	3/7/2001	SUPERCEDED.
	1*	The MPS/Aura simulator shall be capable of maintaining a flag, under operator control, which, when set, shall link startup of CLCW packet transmission to telemetry transmission startup when in IP mode.	3/7/2001	
TLM-20	1	The MPS/Aura simulator shall accept and execute operator directives that start and stop logging of telemetry and CLCWs independently.		
TLM-21	1	The MPS/Aura simulator shall generate EDUs and EDOS data headers based on the User Datagram Protocol (UDP) format defined in the EDOS External ICD Data Format Control Document.		
TLM-22	1	The MPS/Aura simulator shall allow modification of any field within the EDOS data header.		
TLM-23	1	The MPS/Aura simulator shall provide the capability of transmitting the CLCW in the form of EDUs to EOC through EBnet.		
TLM-25	1	The MPS/Aura simulator shall provide for the storage of EDUs during the testing session for later transmission.		
TLM-26	1	The MPS/Aura simulator shall transmit EDUs on an as built basis.		
TLM-27	1	The MPS/Aura simulator shall provide the capability to transmit EDUs using the UDP protocol.		
TLM-28	1	The MPS/Aura simulator interface with the EOC shall comply with the telemetry interface formats and protocols specified in the EDOS to EGS Elements interface document		
TLM-29	1	The MPS/Aura simulator, when acting as a spacecraft, shall comply with the telemetry data formats and protocols specified in the TGT to EDOS interface document.		
TLM-30	1	The MPS/Aura simulator, when acting as an EPGS, shall comply with the telemetry data formats and protocols specified in applicable interface documents for the EPGS to EDOS interface.		
TLM-31	1	The MPS/Aura simulator shall provide the capability to accept Aura telemetry data by electronic transmission and by physical media.		
TLM-32	1	The MPS/Aura simulator shall be capable of transmitting the contents of a user provided file containing Aura telemetry data.		
TLM-34	1	The MPS/Aura simulator shall provide the capability to store up to 8MB of transmitted EDUs.		

Attachment G: Requirements Matrix

MPS/Aura Requirement	Release	MPS/Aura Requirement Description	Date of Insert/Update	Comments
TLM-35	1	The MPS/Aura simulator shall be capable of modifying multiple consecutive buffer locations via a single operator directive.		
TLM-36	1*	The MPS/Aura simulator shall be capable of a limited simulation of clock correlation telemetry. The details of the implementation of this requirement are TBD.	3/7/2001	SUPERCEDED.
	1*	The MPS/Aura simulator shall be capable of a limited simulation of clock correlation telemetry. This simulation shall be limited to populating APID 1000 packets with simulated time and a simulated VCDU sequence counter value, as described in the Aura Spacecraft to EOS Ground System ICD	3/7/2001	Reworded to describe functionality provided. GIIS time, required in the Aqua simulator, has been removed.
TLM-36.1	1	The MPS/Aura simulator shall maintain correlation between the simulated VCDU sequence counter transmitted in APID 1000 packets and in CLCW packets, when operating in IP mode.	3/7/2001	Added to describe functionality provided.
TLM-36.2	1	The MPS/Aura simulator shall maintain correlation between the simulated VCDU sequence counter transmitted in APID 1000 packets and the enclosing CADU, when operating in Serial mode.	3/7/2001	Added to describe functionality provided.
TLM-37	3	The MPS/Aura simulator shall be capable of transmitting telemetry packets over a 1553B interface, employing multiple Remote Terminal numbers and subaddresses.	3/7/2001	See also requirement GEN-21.
TLM	TBD	The MPS/Aura simulator shall be capable of reading telemetry packets from a disk file and interleaving them into the stream of simulator-generated packets.		Proposed.
TLM	TBD	The MPS/Aura simulator shall be capable of simulating an instrument memory dump of loaded data.	3/7/2001	Proposed. The exact capability is TBN.

The requirements defined as To Be Determined (TBD) and/or To Be Negotiated (TBN) are capabilities that may be added to the MPS/Aura simulator. The exact Level 4 requirements will be defined after negotiation of the desired capability.

Attachment H – System Limitations

H.1 MPS/Aura Release 2.0 Limitations

The following limitations apply to MPS/Aura Release 2.0. Some of these are Discrepancy Reports (DRs) against SIMSS baseline products and have been recorded in their DR repository.

Problem Description	Workaround
The event message window can accept no more than 50 messages per second. The Scenario module can easily overrun this limit and flood the GUI with messages.	Use Sleep directives to slow scenario files to 50 directives per second or less.
The Generic Container Buffer display is limited to 1400 bytes of data (= 700 words, or 350 double words). A request for more data than that will result in a display of 1400 bytes of information. *This is SIMSS Defect # 102.*	To view data that is beyond byte 1400 of the buffer, set the offset to 1400, or as required to view the data.
If a container item name such as a telemetry mnemonic is entered into multiple displays of the *Display/Set Container Items...* window and updated in a higher numbered display, the update will not be reflected in lower numbered display(s).	Do not duplicate container item names.
When using SQL*Plus to select entries from the Oracle calcurve table via the conversion type field, *conv_type*, it is necessary to put a space after the type entry. e.g. "U_5D ", not "U_5D".	Given at left.

The total of all while loops in scenario scripts running at any given time cannot exceed 1000.	Construct while loops so that the total never exceeds 1000 iterations. If the script must loop for more than that, exit the loop after 1000 iterations and use the start scenario directive to call the scenario file again. Ex.:
	while (I < 1000)
	    set <mnemonic> += 1
	    set I += 1
	    sleep 500
	endwhile
	start scenario <my_own_name>

Attachment I - Release History Summary Matrix

Attached is the release history summary matrix, which reflects the MPS/Aura Release 2.0 delivery. Modules inherited from the SIMSS baseline have the SIMSS Release Number, while the MPS/Aura modules EOSGS and SCAURA have the current MPS/Aura Release Number.

Release History Summary Matrix

System: **MPS/Aura**

Release Number		1.0	2.0											
Delivery Date		3/16/01	6/15/01											
Configuration Item	CI No.													
Core (Client)	1.1	4.0	4.1											
Core (Server)	1.2	4.0	4.1											
SCAURA (Client)	1.3	1.0	2.0											
SCAURA (Server)	1.4	1.0	2.0											
EOSGS (Client)	1.5	1.0	2.0											
EOSGS (Server)	1.6	1.0	2.0											
IP Input (Client)	1.7	4.0	4.1											
IP Input (Server)	1.8	4.0	4.1											
IP Output (Client)	1.9	4.0	4.1											
IP Output (Server)	2.0	4.0	4.1											
Logging (Client)	2.1	4.0	4.1											
Logging (Server)	2.2	4.0	4.1											

Delivery Date		3/16/01	6/15/01											
Configuration Item	CI No.													
Scenario (Client)	2.3	4.0	4.1											
Scenario (Server)	2.4	4.0	4.1											
Serial Input (Client)	2.5	4.0	4.1											
Serial Input (Server)	2.6	4.0	4.1											
Serial Output (Client)	2.7	4.0	4.1											
Serial Output (Server)	2.8	4.0	4.1											
TxFile (Client)	2.9	4.0	4.1											
TxFile (Server)	3.0	4.0	4.1											
vcProcessor (Client)[1]	3.1		4.1											
vcProcessor (Server)[1]	3.2		4.1											

---

[1] Added with Release 2.0

<u>**Attachment J - Delivery Details**</u>

J.1 Software

A complete listing of the MPS/Aura software file names is available upon request.

J.2 Hardware for MPS/Aura

One new PC is being delivered to Denver for MPS/Aura. The existing PCs on which the MPS/Aqua simulator runs may also be used to run the MPS/Aura simulator.

J.2.1 Hardware for MPS/Aura units in GSFC, Building 32

There are two CSOC-owned PCs presently installed in Building 32.

PC #1:

Qty	Common Name	Model [Serial No.]	Mfg	CSOC No.	Description
1	Computer	E-4200 001-343-8943	Gateway	C0060047	Intel Pentium II 400 Mhz w /512 Cache, 256 MB SDRAM PC100 6ns Micron, Matrox Millenium II 8MB AGP Video card, Toshiba 32x SCSI CD ROM Drive, Seagate 9.1 GB hard disk, IOMEGA 100 mb internal zip drive
1	Monitor	VX1100 811053233	Gateway	C0060041	21" Monitor
1	Mouse	Intellimouse 2570734-10000	Gateway		
1	Keyboard	Q9045A1837	Gateway		
1	Timing Card	PCIDCC20-P	Industrial Computer Source		PCI counter/timer card
2	Serial I/O Cards	97B1423 97B1424			Mfg name is FASTCOMESCC/P

PC #2:

Qty	Common Name	Model [Serial No.]	Mfg	CSOC No.	Description
1	Computer	E-4200 001-343-8944	Gateway	C0060050	Intel Pentium III 450 Mhz w /512 Cache, 256 MB SDRAM PC100 6ns Micron, Matrox Millenium II 8MB AGP Video card, Toshiba 32x SCSI CD ROM Drive, Seagate 9.1 GB hard disk, IOMEGA 100 mb internal zip drive
1	Monitor	VX1100 811053233	Gateway	C0060041	21" Monitor
1	Mouse	Intellimouse	Gateway		
1	Keyboard		Gateway		
2	Serial I/O Cards	97B1428 97B1429			Mfg name is FASTCOMESCC/P

J.2.2 Hardware for MPS/Aura at Denver

There is one CSOC-owned PC currently installed at Denver.

Qty	Common Name	Model [Serial No.]	Mfg	CSOC No.	Description
1	Computer	E-4200 001-343-8946	Gateway	C0060052	Intel Pentium III 450 Mhz w /512 Cache, 256 MB SDRAM PC100 6ns Micron, Matrox Millenium II 8MB AGP Video card, Toshiba 32x SCSI CD ROM Drive, Seagate 9.1 GB hard disk, IOMEGA 100 mb internal zip drive
1	Monitor	VX1100 811053230	Gateway	C0060043	21" Monitor
1	Mouse	Intellimouse	Gateway		
1	Keyboard		Gateway		

Specifications for the new PC being delivered to Denver:

Qty	Common Name	Model [Serial No.]	Mfg	CSOC No.	Description
1	Computer	E-4200 001-343-8945	Gateway	C0060048	Intel Pentium III 450 Mhz w /512 Cache, 256 MB SDRAM PC100 6ns Micron, Matrox Millenium II 8MB AGP Video card, Toshiba 32x SCSI CD ROM Drive, Seagate 9.1 GB hard disk, IOMEGA 100 mb internal zip drive
1	Monitor	VX1100 811053463	Gateway	C0060044	21" Monitor
1	Mouse	Intellimouse	Gateway		
1	Keyboard		Gateway		

Attachment K - Documentation References

The following documents have been employed as the main sources for direction and information in producing Release 2.0 of the MPS/Aura simulator. The Data Format Control Document (DFCD) for Aura is not yet available so the PM-1 (Aqua) DFCD, which is understood to be very similar, was used.

Document	Location*
Earth Observing System (EOS) Common Spacecraft Program Interface Control Document between the EOS Aura Spacecraft and the EOS Ground System, Dated April 15, 2000, Document No. D27515, Preliminary (more commonly known as "The Space to Ground ICD")	1
Data Format Control Document for the Earth Observing System (EOS) Mission Operations Segment (EMOS) Project Database Volume 1: PM-1 Users Revision -, dated January 2000	4
TRW, EOS Chemistry Spacecraft Flight Software Requirements Specification, ES-SDA-005, Rev. A, dated 23 August, 2000	1
TRW, EOS Aura Spacecraft Flight Software User's Guide, No.: D31189, version dated 15 September, 2000	1
TRW, EOS Aura Command Allocation Document, No.: D31174, dated July 27, 2000	1
TRW, EOS Aura Telemetry Allocation Document, No.: D31175, dated September 22, 2000	1
TRW, EOS PM-1 Spacecraft Equipment Specification for Transponder Interface Electronics, No.: EQ4-4957, latest version dated 11 February, 1999	1
TRW, Interface Control Document Between the Earth Observing System (EOS) Data and Operations System (EDOS) and the EOS Ground System (EGS) Elements CDRL B301	2
Consultative Committee for Space Data Systems, CCSDS 102.0-B-4: Packet Telemetry Blue Book, Issue 4, Nov. 1995	3
--, CCSDS 202.1-B-1: Telecommand Part 2.1 – Command Operations Procedures Blue Book, Issue 1, Oct. 1995	3
NASA, GSFC, Earth Observing System Data and Information System (EOSDIS) Test System (ETS) Level 4 Requirements for the MPS/Aura. Version listed in Attachment G.	This package.

*Location Legend:

Number	Designation
1	TRW web server (URL not listed for security reasons)
2	http://esdis-it.gsfc.nasa.gov:8080/servlet/DOC?nDocindex=14
3	http://ccsds.org/publications.html
4	ftp://198.118.192.20/pub/fot/pm1/pdb/DFCD/
5	http://esdis-it.gsfc.nasa.gov/ETS/etsdoc.html

Attachment L — Mission Systems Configuration Management Form

This attachment contains the completed Mission Systems Configuration Management (MSCM) form for the delivery of MPS/Aura Release 2.0.

Mission Systems Configuration Management Form

1. ORIGINATOR	2. ORGANIZATION	3. PHONE	4. E-MAIL ADDRESS
Dave Green	CSC	301-805-3420	dsgreen@csc.com

5. ELEMENT	6. INSTALLATION PRIORITY	7. TRACKING NUMBER
ETS (MPS/Aura)	Routine	(Assigned by CM Office)

8. SOURCE CHANGE REQUEST(S):	9. APPROVALS		
ETS delivery of MPS for EOS Aura (MPS/Aura)	Element Manager	_____	___/___/___
	Flight Ops Director	_____	___/___/___
	Operations Manager	_____	___/___/___

10. DELIVERED SYSTEM (Check all that apply)

	Name	Version	Media Identification	Identification Date
☐ Hardware				
☒ Software	MPS/Aura	R2.0	CD-ROM	06/15/01
☐ Database				
☒ Documentation:				
MPS/Aura delivery package	N/A	3.5 " Diskette	06/15/01
MPS/Aura Release 2.0 User's Guide	R2.0	http://esdis-it.gsfc.nasa.gov/ETS/etsdoc.html	TBS
☐ Other				

11. CHANGE DESCRIPTION

Release 2.0 of MPS/Aura (MPS/Aura)

12. ATTACHMENT(S): Check if YES ☒

Description: MPS/Aura Release 2.0 delivery package (cover letter with attachments) dated 06/15/01

13. CM OFFICE USE

	Location (Bldg/Room)	Slot location(s)
Hardware	_____/_____	_____
Media	_____/_____	_____
Documentation	_____/_____	_____
Installation date	___/___/___	CM Office Signature _____

Form MSCM (970327)